CLAIMS

WHAT IS CLAIMED IS:

1. A method for packet routing, comprising:

receiving a packet that includes an address;

fetching a first stride block based on a first portion of the address, wherein each stride block includes a first bitmap, a second bitmap, a leaf table base pointer, and a branch table base pointer;

processing the first stride block, wherein processing a stride block includes:

determining if a forwarding decision is determined based on a second portion of the address and at least one of the first and second bitmaps of the stride block;

when the forwarding decision is determined based on at least one of the first and second bitmaps:

generating a leaf table offset from at least one of the first and second bitmaps and the second portion of the address;

combining the leaf table offset with the leaf table base pointer to produce a leaf table index; and

accessing a leaf table using the leaf table index to retrieve the forwarding decision;

when the forwarding decision is not determined based on the second portion of the address and at least one of the first and second bitmaps;

generating a branch table offset from the second portion of the address and at least one of the first and second bitmaps;

combining the branch table offset with the branch table base pointer to produce a branch table index;

accessing a branch table using the branch table index to retrieve a subsequent stride block; and

processing the subsequent stride block and any additional subsequent stride blocks generated using additional portions of the address until the forwarding decision is retrieved.

2. The method of claim 1, wherein the first bitmap is a leaf bitmap and the second bitmap is a branch bitmap, wherein determining if a forwarding decision is determined further comprises determining if the forwarding decision is determined based on the second portion of the address and the leaf and branch bitmaps, wherein generating the leaf table offset further comprises generating the leaf table offset from the second portion of the address and the leaf bitmap, and wherein generating the branch table offset further comprises generating the branch table offset from the second portion of the address and the branch bitmap.

3. The method of claim 2, wherein generating the leaf table offset further comprises:

masking off a portion of the leaf bitmap to produce a masked leaf bitmap, wherein the portion of the leaf bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked leaf bitmap to produce the leaf offset.

4. The method of claim 2, wherein generating the branch table offset further comprises:

masking off a portion of the branch bitmap to produce a masked branch bitmap, wherein the portion of the branch bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked branch bitmap to produce the branch offset.

5. The method of claim 1, wherein accessing the leaf table using the leaf table index to retrieve the forwarding decision further comprises:

accessing the leaf table to retrieve a pointer to the forwarding decision; and

retrieving the forwarding decision using the pointer to the forwarding decision.

6. The method of claim 1, wherein the first bitmap is an extends bitmap and the second bitmap is a type bitmap, wherein determining if a forwarding decision is determined further comprises determining if the forwarding decision is determined based on the second portion of the address and the extends bitmap, wherein generating the leaf table offset further comprises generating the leaf table offset from the second portion of the address and the extends and type bitmaps,

and wherein generating the branch table offset further comprises generating the branch table offset from the second portion of the address and the extends and type bitmaps.

7. The method of claim 6, wherein generating the leaf table offset further comprises:

combining the extends bitmap and the type bitmap to generate a leaf bitmap;

masking off a portion of the leaf bitmap to produce a masked leaf bitmap, wherein the portion of the leaf bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked leaf bitmap to produce the leaf offset.

8. The method of claim 6, wherein generating the branch table offset further comprises:

combining the extends bitmap and the type bitmap to generate a branch bitmap;

masking off a portion of the branch bitmap to produce a masked branch bitmap, wherein the portion of the branch bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked branch bitmap to produce the branch offset.

9. The method of claim 8, wherein the type bitmap identifies each subsequent stride block as one of a sparse stride block and a dense stride block.

10. The method of claim 9, wherein subsequent stride blocks for each stride block are stored in contiguous sets.

11. The method of claim 10, wherein sparse blocks are grouped together and dense blocks are grouped together in the contiguous sets.

12. A packet routing processor, comprising:

a processing module; and

memory operably coupled to the processing module, wherein the memory stores operating instructions such that, when executed by the processing module, the operating instructions cause the processing module to perform the functions of:

fetching a first stride block based on a first portion of an address for a received packet, wherein each stride block includes a first bitmap, a second bitmap, a leaf table base pointer, and a branch table base pointer;

processing the first stride block, wherein processing a stride block includes:

determining if a forwarding decision is determined based on a second portion of the address and at least one of the first and second bitmaps of the stride block;

when the forwarding decision is determined based on at least one of the first and second bitmaps:

generating a leaf table offset from at least one of the first and second bitmaps and the second portion of the address;

combining the leaf table offset with the leaf table base pointer to produce a leaf table index; and

accessing a leaf table using the leaf table index to retrieve the forwarding decision;

when the forwarding decision is not determined based on the second portion of the address and at least one of the first and second bitmaps;

generating a branch table offset from the second portion of the address and at least one of the first and second bitmaps;

combining the branch table offset with the branch table base pointer to produce a branch table index;

accessing a branch table using the branch table index to retrieve a subsequent stride block; and

processing the subsequent stride block and any additional subsequent stride blocks generated using additional portions of the address until the forwarding decision is retrieved.

13. The packet routing processor of claim 12, wherein the first bitmap is a leaf bitmap and the second bitmap is a branch bitmap, wherein determining if a forwarding decision is determined further comprises determining if the forwarding decision is determined based on the second portion of the address and the leaf and branch bitmaps, wherein generating the leaf table offset further comprises generating the leaf table offset from the second portion of the address and the leaf bitmap, and wherein generating the branch table offset further comprises generating the branch table offset from the second portion of the address and the branch bitmap.

14. The packet routing processor of claim 13, wherein generating the leaf table offset further comprises:

masking off a portion of the leaf bitmap to produce a masked leaf bitmap, wherein the portion of the leaf bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked leaf bitmap to produce the leaf offset.

15. The packet routing processor of claim 13, wherein generating the branch table offset further comprises:

masking off a portion of the branch bitmap to produce a masked branch bitmap, wherein the portion of the branch bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked branch bitmap to produce the branch offset.

16. The packet routing processor of claim 12, wherein accessing the leaf table using the leaf table index to retrieve the forwarding decision further comprises:

accessing the leaf table to retrieve a pointer to the forwarding decision; and

retrieving the forwarding decision using the pointer to the forwarding decision.

17. The packet routing processor of claim 12, wherein the first bitmap is an extends bitmap and the second bitmap is a type bitmap, wherein determining if a forwarding decision is determined further comprises determining if the

forwarding decision is determined based on the second portion of the address and the extends bitmap, wherein generating the leaf table offset further comprises generating the leaf table offset from the second portion of the address and the extends and type bitmaps, and wherein generating the branch table offset further comprises generating the branch table offset from the second portion of the address and the extends and type bitmaps.

18. The packet routing processor of claim 17, wherein generating the leaf table offset further comprises:

combining the extends bitmap and the type bitmap to generate a leaf bitmap;

masking off a portion of the leaf bitmap to produce a masked leaf bitmap, wherein the portion of the leaf bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked leaf bitmap to produce the leaf offset.

19. The packet routing processor of claim 17, wherein generating the branch table offset further comprises:

combining the extends bitmap and the type bitmap to generate a branch bitmap;

masking off a portion of the branch bitmap to produce a masked branch bitmap, wherein the portion of the branch bitmap that is masked off is determined based on the second portion of the address; and

performing a population count on the masked branch bitmap to produce the branch offset.

20. The packet routing processor of claim 19, wherein the type bitmap identifies each subsequent stride block as one of a sparse stride block and a dense stride block.

21. The packet routing processor of claim 20, wherein dense stride blocks store subsequent sparse stride blocks and subsequent dense stride blocks in contiguous sets.

22. A packet routing circuit, comprising:

a packet memory receives packets and stores the packets prior to forwarding, wherein each packet includes an address;

output circuitry operably coupled to the packet memory, wherein for each packet, the output circuitry receives a forwarding decision and forwards the packet to at least one of a plurality of outputs based on the forwarding decision;

memory that stores a forwarding table, wherein the forwarding table stores forwarding decisions for the packets; and

a determination block operably coupled to the memory and the output circuitry, wherein the determination block receives the address for each packet and determines the forwarding decision for the packet, wherein the determination block provides the forwarding decision for the packet to the output circuitry such that the packet is forwarded to at least one of the outputs, wherein determining the forwarding decision for the packet includes:

fetching a first stride block from the forwarding table stored in the memory based on a first portion of an address for a received packet, wherein each stride block includes a first bitmap, a second bitmap, a leaf table base pointer, and a branch table base pointer;

processing the first stride block, wherein processing a stride block includes:

determining if a forwarding decision is determined based on a second portion of the address and at least one of the first and second bitmaps of the stride block;

when the forwarding decision is determined based on at least one of the first and second bitmaps:

generating a leaf table offset from at least one of the first and second bitmaps and the second portion of the address;

combining the leaf table offset with the leaf table base pointer to produce a leaf table index; and

accessing a leaf table stored in the memory using the leaf table index to retrieve the forwarding decision;

when the forwarding decision is not determined based on the second portion of the address and at least one of the first and second bitmaps;

generating a branch table offset from the second portion of the address and at least one of the first and second bitmaps;

combining the branch table offset with the branch table base pointer to produce a branch table index;

accessing a branch table stored in the memory using the branch table index to retrieve a subsequent stride block; and

processing the subsequent stride block and any additional subsequent stride blocks generated using additional portions of the address until the forwarding decision is retrieved.

23. The packet routing circuit of claim 22 further comprises a cache operably coupled to the memory and the determination block, wherein the cache stores at least a portion of the forwarding table.

24. The packet routing circuit of claim 22, wherein the memory stores a plurality of forwarding tables, wherein a particular forwarding table is selected for use in determining the forwarding decision for a particular packet based on at least one of a field included in the particular packet and an identity of an input port to the packet routing circuit over which the particular packet was received.

25. The packet routing circuit of claim 22, wherein the determination block further comprises:

a processing module; and

an instruction memory operably coupled to the processing module, wherein the instruction memory stores instructions that, when executed by the processing module, cause the processing module to perform functions necessary to determine the forwarding decision for the packet.

26. The packet routing circuit of claim 22, wherein the determination block further comprises a state machine.

27. The packet routing circuitry of claim 22, wherein the packet routing circuitry is included in a router.

28. The packet routing circuit of claim 27, wherein the packets are internet protocol (IP) packets.

29. The packet routing circuit of claim 22, wherein the determination block utilizes population counts to determine branch and leaf offsets.

30. The packet routing circuit of claim 22, wherein the determination block utilizes linear operations to determine the forwarding decision for each of the packets.

31. A method for compressing a stride included in a trie structure, wherein the stride includes a plurality of nodes, comprising:

separating the stride into a plurality of stride portions, wherein each stride portion includes stride results for a portion of the plurality of nodes, wherein a stride result is one of a leaf pointer and a branch pointer; and

for each stride portion:

compressing the stride results for the stride portion using run length encoding to produce a compression bitmap and a compressed list of stride results;

generating a leaf bitmap, a branch bitmap, a leaf table section, and a branch table section from the compression bitmap and the compressed list of stride results;

storing the leaf table section in a leaf table at a leaf table base pointer for the stride portion;

storing the branch table section in a branch table at a branch table base pointer for the stride portion; and

storing a stride block in memory for the stride portion, wherein the stride block includes the leaf bitmap, the branch bitmap, the leaf table base pointer, and the branch table base pointer.

32. The method of claim 31, wherein compressing the results for a stride block further comprises selecting one of a sparse compression format and a dense compression format based on a number of compressed stride results; and wherein

storing the stride block in memory further comprises storing the stride block in the selected one of the sparse compression format and the dense compression format.

33. The method of claim 32, wherein the leaf bitmap and branch bitmap for each stride block are encoded in an extends bitmap and a type bitmap, wherein the extends bitmap and type bitmap for each stride block encode sparse and dense format distinctions for stride blocks accessed via branch pointers included in the branch table.

34. The method of claim 33, wherein the sparse compression format includes a value array corresponding to values for address bits used to access the stride block, and wherein the dense compression format includes bitmaps corresponding to the address bits, wherein a quantity of memory required to store the stride block in the sparse compression format is less than a quantity of memory required to store the stride block in a dense compression format.

35. A method for packet routing, comprising:

receiving a packet that includes an address;

fetching a first stride block, wherein the first stride block encodes a first portion of a longest prefix match trie, wherein the first stride block is one of a sparse stride block and a dense stride block, wherein sparse stride blocks encode portions of the longest prefix match trie that include no more than a first number of nodes, wherein dense stride blocks encode portions of the longest prefix match trie that include more than the first number of nodes;

comparing a first portion of the address with a first portion of the first stride block to determine if a forwarding decision for the packet is resolved by the first stride block;

when the forwarding decision is resolved by the first stride block, determining the forwarding decision for the packet;

when the forwarding decision for the packet is not resolved by the first stride block:

determining a second stride block based on the first portion of the address and a second portion of the first stride block; and

processing the second stride block and any subsequent stride blocks determined until the forwarding decision is determined, wherein processing a stride block includes fetching the stride block and comparing a portion of the address with a portion of the stride block to determine one of the forwarding decision and a subsequent stride block for processing; and

forwarding the packet based on the forwarding decision.

36. The method of claim 35, wherein when the first stride block is a dense stride block, comparing a first portion of the address with the portion of the first stride block further comprises:

selecting a stride record of a plurality of stride records included in the first stride block using a first set of bits in the first portion of the address, wherein each stride record of the plurality of stride records encodes a portion of nodes encoded by the first stride block; and

comparing a second set of bits in the first portion of the address with a portion of the stride record to determine if the forwarding decision is resolved by the first stride block.